

RECHERCHE D'IMAGES COULEUR À PARTIR DE LEURS HISTOGRAMMES

par

Bilal Tawbe

mémoire présenté au Département de mathématiques
et d'informatique en vue de l'obtention du grade de maître ès sciences (M.Sc.)

**FACULTÉ DES SCIENCES
UNIVERSITÉ DE SHERBROOKE**

Sherbrooke, Québec, Canada, février 1999



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*

Our file *Notre référence*

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-56974-8

Canada

Le 16/2/2020, le jury suivant a accepté ce mémoire dans sa version finale.
date

Président-rapporteur: M.Froduald Kabanza
Département de mathématiques et d'informatique

Membre: M. Djemel Ziou
Département de mathématiques et d'informatique

Membre: M.Béchir Ayeb
Département de mathématiques et d'informatique

Membre: M. Shengrui Wang
Département de mathématiques et d'informatique

À mes parents,

*À mon père, ma mère et mes frères,
à toute ma famille,
avec toute mon affection.*

SOMMAIRE

Le but de ce mémoire est de construire un système très évolué capable de détecter les images semblables à une image donnée. L'utilisateur choisit ou construit une image-requête, et recherche dans la base d'images disponibles celles qui ressemblent à l'image-requête. Le système indexe les images par des techniques automatiques d'analyse d'images s'appuyant sur les couleurs. Chaque image couleur est représentée par un histogramme couleur (chaque histogramme couleur contient trois histogrammes qui représentent les couleurs rouge, vert et bleu). Nous aborderons plusieurs problèmes, dont les suivants :

- Comment peut-on extraire des images similaires?
- Comment peut-on construire une base de données d'images couleur, simple d'utilisation?

Pour chercher les images semblables à une image donnée, nous avons décrit deux méthodes efficaces, l'une parcourant toute une base de données, l'autre repérant des centres trouvés par l'algorithme SOFM (Self-Organizing Feature-Mapping). Ce dernier nous permet d'éviter une recherche dans toute la base de données et d'accélérer la vitesse d'exécution. Enfin nous avons fait une comparaison entre les résultats obtenus par les deux méthodes pour connaître la moyenne du taux de réussite relative à chacune.

REMERCIEMENTS

Je tiens d'abord à remercier sincèrement mon père et ma mère qui m'ont encouragé à poursuivre mes études.

Je voudrais exprimer ma plus profonde reconnaissance à mon directeur de recherche, M. Shengrui Wang, ainsi qu'à mon codirecteur, M. Djemel Ziou, professeurs au département de mathématiques et d'informatique de l'Université de Sherbrooke, pour leur attention, leur dévouement et leur soutien tout au long de ma recherche et de ma rédaction. Leur aide et leurs conseils m'ont aidé à organiser et à éclaircir ce travail.

Je remercie sincèrement le gouvernement du Québec pour l'aide financière qu'il m'a accordée au cours de ma maîtrise.

Enfin, je remercie tous mes amis, qui ont rendu mon travail plus aisé par leurs commentaires.

TABLE DES MATIÈRES

SOMMAIRE	iii
REMERCIEMENTS	iv
TABLE DES MATIÈRES	v
LISTE DES TABLEAUX	vii
LISTE DES FIGURES	viii
INTRODUCTION	1
1 Recherche des images similaires à partir de leurs histogrammes couleur	4
1.1 Base d'images	5
1.2 Histogramme couleur et calcul de la distance	7
1.2.1 Histogramme couleur	7
1.2.2 Calcul de la distance	8
1.2.3 Espace d'histogrammes	10
1.2.4 Distribution des distances	11
1.2.5 Capacité de l'espace d'histogrammes	12
1.3 Les paramètres d'optimisation du temps de recherche	14

1.3.1	Probabilité de similitude et relation avec le nombre moyen des images similaires	15
1.3.2	Calcul du seuil t	16
1.3.3	Estimation de la probabilité	16
1.4	Algorithmes	18
1.4.1	Algorithme 1 : Indexation de la base de données	18
1.4.2	Algorithme 2 : Recherche des images similaires à une image donnée	19
1.4.3	Algorithme 3 : Calcul du seuil qui contrôle le nombre moyen d'images semblables	20
1.4.4	Algorithme 4 : Calcul de la probabilité de similitude entre deux histogrammes	22
1.5	Résultats	22
1.6	Conclusion	31
2	Indexation à partir des réseaux neuronaux SOFM	33
2.1	Fonctionnement et apprentissage	36
2.2	Réseaux SOFM	37
2.3	Nouvelle indexation de la base d'images	39
2.4	Résultats	41
	CONCLUSION	46
	BIBLIOGRAPHIE	48

LISTE DES TABLEAUX

1	Valeurs de la capacité à un t donné.	14
2	Valeurs de la probabilité à chaque t donnée.	15
3	Moyennes des images semblables pour 511 images, pour chaque valeur de R donnée par l'utilisateur.	30
4	Moyennes des taux de réussite, pour 511 images, pour chaque valeur de R donnée par l'utilisateur.	45

LISTE DES FIGURES

1	Schéma général du système de recherche.	6
2	a) Image couleur et b) son histogramme normalisé	9
3	Distribution des distances.	12
4	La probabilité et son estimation.	18
5	(a) Image originale. La première image (b) est l'image dont la distance à l'image originale est nulle. Le nombre moyen d'images $R = 5$, $t = 170.9$ et $tr = 87s$	24
6	La distance entre les histogrammes (a) et (b) est nulle et la distance entre les histogrammes (a) et (c) est petite.	25
7	Trois images similaires à l'image originale. Le nombre moyen d'images $R = 5$, $t = 170.9$ et $tr = 40s$. Le seuil t dépend de R	26
8	(a) Histogramme de l'image originale. (b), (c) et (d) sont les histogrammes des images similaires à l'image originale. Les histogrammes se ressemblent mais les images pas nécessairement.	27
9	Quatre images similaires à l'image originale. Le nombre moyen d'images $R = 5$, $t = 170.9$ et $tr = 30s$	28
10	(a) Image originale. Nous avons trouvé cinq images similaires à l'image originale. Le nombre moyen d'images semblables $R = 5$, $t = 170.9$ et $tr = 44s$	29

11	L'image originale est dans (a). Nous avons trouvé cinq images similaires à cette image. L'ordre des images similaires est relatif au degré de similitude décroissant avec l'image originale. $R = 5$, $t = 170.9$ et $tr = 45s$	30
12	Système de recherche à partir des centres.	35
13	L'image originale est dans (a). Nous avons trouvé deux images similaires à celle-ci. Le nombre moyen d'images $R = 5$ et le seuil $t = 206.6$	42
14	L'image originale est dans (a). Deux images similaires à cette image ont été identifiées. La première image (b) est la plus semblable à l'image originale, $R = 5$ et $t = 206.6$	43
15	Trois images similaires à l'image originale, $R = 5$ et $t = 206.6$	44

INTRODUCTION

La recherche des images par contenu est très utile dans plusieurs domaines comme la transmission et le stockage d'images, la synthèse d'images, l'imagerie par satellite, les ressources terrestres, la météorologie, la télévision, les communications militaires et l'imagerie médicale. La structure complexe des images demande de nouvelles techniques pour les représenter et les extraire à partir d'une base de données très large. La recherche traditionnelle à partir d'un texte ou d'un mot-clé reste limitée et ne peut pas être utilisée dans toutes les applications. Ce texte ou mot-clé sont générés manuellement. Cependant, cette description n'est pas suffisamment riche pour détecter le contenu visuel de l'image.

La recherche d'images par "le contenu" consiste à développer des outils permettant de sélectionner les images les plus pertinentes par leur contenu, comme c'est le cas pour les systèmes de recherche d'informations. Dans le cas général d'une recherche d'images par similitude (recherche avec modèle), l'utilisateur choisit ou construit une image-requête, et recherche dans la base des images disponibles celles qui ressemblent à l'image-requête.

Dans notre travail, nous avons utilisé deux méthodes pour rechercher les images semblables à une image donnée. La première méthode consiste à représenter chaque image couleur par un vecteur histogramme et à le comparer avec tous les vecteurs associés aux images de la base de données. Dans la deuxième méthode, nous avons utilisé un modèle de réseaux neuronaux pour éviter la recherche dans toute la base de données.

L'objectif de notre travail est le suivant :

- construire et indexer une base de données d'images simple à utiliser, répondant aux

besoins des utilisateurs ;

- calculer les paramètres nécessaires à cette construction ;
- élaborer des algorithmes pouvant trouver toutes les images similaires à une image donnée.

Pour construire une base de données, il est nécessaire d'indexer les images sous une forme simple d'utilisation. Pour rechercher les images similaires à une image donnée, nous calculons la distance entre l'histogramme représentant cette image donnée et tous les histogrammes associés aux images de la base de données [1]. Nous extrayons toutes les images dont la distance est inférieure ou égale à un seuil que nous avons calculé. Plusieurs recherches réalisées ont mené à de bons résultats ; Yihong Gong, Chua Hock et Guo Xiaoyi (1996) [1] ont proposé une nouvelle méthode pour créer un histogramme insensible à la variation de la lumière. Markus Stricker et Michael Swain (1994) [2] et Stricker (1992) [5] ont utilisé une méthode basée sur l'histogramme couleur. Funt et Finlayson (1991) [6] et Swain et Ballard (1991) [7] ont utilisé une base de données contenant moins de cent images. Ils ont utilisé différentes sortes d'histogrammes. Engelson et McDermont (1991) [8] ont utilisé 168 images pour tester leurs algorithmes, ainsi que des histogrammes relatifs aux propriétés des textures. Dans notre travail, les algorithmes sont testés sur 511 images et nous avons utilisé des histogrammes de couleurs. L'image donnée est comparée avec des images représentatives trouvées à partir des réseaux neuronaux SOFM.

Le premier chapitre aborde le processus d'indexation (quelques notions générales sur les images couleur sont données), l'élaboration d'une base de données et la technique de représentation des images couleur par des histogrammes. Chaque histogramme comporte trois composantes représentant les couleurs. Nous avons utilisé une distance métrique entre les vecteurs pour trouver les images similaires à une image donnée. Nous avons déterminé la probabilité de similitude de deux histogrammes pour calculer le seuil. Nous avons explicité une méthode très efficace pour calculer un seuil particulier, et l'avons utilisée pour préciser le nombre d'images similaires désirées. Par exemple, si l'utilisateur

désire dix images, on peut lui donner approximativement les dix images les plus proches à partir du seuil calculé. Les algorithmes choisis pour faire l'indexation, le calcul du seuil et l'extraction des images similaires à une image donnée ont été décrits. Nous avons établi la relation entre la probabilité et le nombre moyen d'images dans la base de données. Les résultats obtenus dans cette partie sont appréciables.

Dans le deuxième chapitre, nous exposons une méthode importante pour faire l'indexation à partir des réseaux neuronaux. Nous avons utilisé les réseaux SOFM pour calculer plusieurs centres et pour extraire les nuages liés à ces centres. Le nuage d'un centre est l'ensemble des images relatives à ce centre. Nous avons conçu de nouveaux algorithmes pour faire une nouvelle indexation à partir des réseaux SOFM et pour chercher les images similaires à l'image donnée. L'avantage de ces nouveaux algorithmes est d'éviter la recherche dans toute la base de données. L'image donnée est comparée avec les centres calculés pour enfin extraire les images similaires à cette image.

Enfin, nous présentons les résultats que nous avons obtenus concernant les images en couleurs et la recherche des images similaires à une image donnée. Les portes sont donc ouvertes et offrent la chance d'améliorer la recherche d'images couleur dans une base de données.

Chapitre 1

Recherche des images similaires à partir de leurs histogrammes couleur

Dans ce chapitre nous expliquerons la première méthode que nous avons utilisée. Dans cette méthode nous comparons l'image donnée avec toutes les images de la base de données. Nous avons élaboré une base de données de 511 images couleur dans le format JPEG ; chaque image y est représentée par son histogramme. Nous créons un index dans la base de données contenant 511 histogrammes où chaque histogramme représente une image de la base de données. Pour une image requête quelconque incluse ou non dans la base de données, nous calculons d'abord son histogramme. Pour extraire les images similaires à l'image requête, il faut calculer les distances entre leur histogramme et les histogrammes indexés dans la base de données. Nous extrayons les images dont la distance est inférieure ou égale à un seuil déterminé par calcul. Dans ce chapitre nous aborderons plusieurs problèmes intéressants ; parmi ceux-ci nous avons :

- Comment représenter des images couleur par des histogrammes ?
- Comment construire une base de données d'images couleur simple d'utilisation et répondant aux besoins des utilisateurs ?
- Comment établir la relation entre le nombre moyen des images similaires à une

image (requête) et la probabilité de similitude entre deux images?

Le schéma (figure 1) explique les étapes nécessaires pour construire une base d'images et pour rechercher les images similaires à une image donnée. Chaque image de la base est représentée par un histogramme; soit I une image donnée et soit H son histogramme couleur calculé. Nous calculons les distances entre H et tous les histogrammes représentant les images dans la base de données. Calculons la probabilité de similitude entre deux histogrammes pour obtenir la valeur d'un certain seuil désiré t , afin de détecter un nombre moyen précis des images similaires. Enfin extrayons les images dont la distance à l'image I est inférieure ou égale à t . Dans ce chapitre nous allons expliquer ces étapes et les résultats obtenus. Nous allons présenter la base de données que nous avons construite, la représentation par histogramme couleur et le calcul de la distance entre deux histogrammes. Dans la section 1.2.3, 1.2.4 et 1.2.5 nous allons étudier successivement l'espace d'histogrammes, la distribution des distances des histogrammes et la capacité de cet espace. La section 1.4 explique comment faire l'indexation de notre base de données. La section 1.5 explique l'extraction des images semblables à une image donnée. Le calcul de la probabilité de similitude entre deux histogrammes et la relation entre cette probabilité et le nombre moyen d'images sont expliqués en détail dans la section 1.5.

1.1 Base d'images

Nous avons constitué une base de données de 511 images couleur de type photographique. Toutes ces images sont prises au Liban dans des environnements différents. Elles sont stockées en format JPEG. Ce format est utilisé pour supprimer certaines couleurs dans une image, invisibles à l'oeil nu, son utilisation étant destinée à des images de type photographique. Chaque point d'une image peut avoir trois différentes couleurs. Les images de la base de données sont numérotées par ordre croissant de 1 jusqu'à 511.

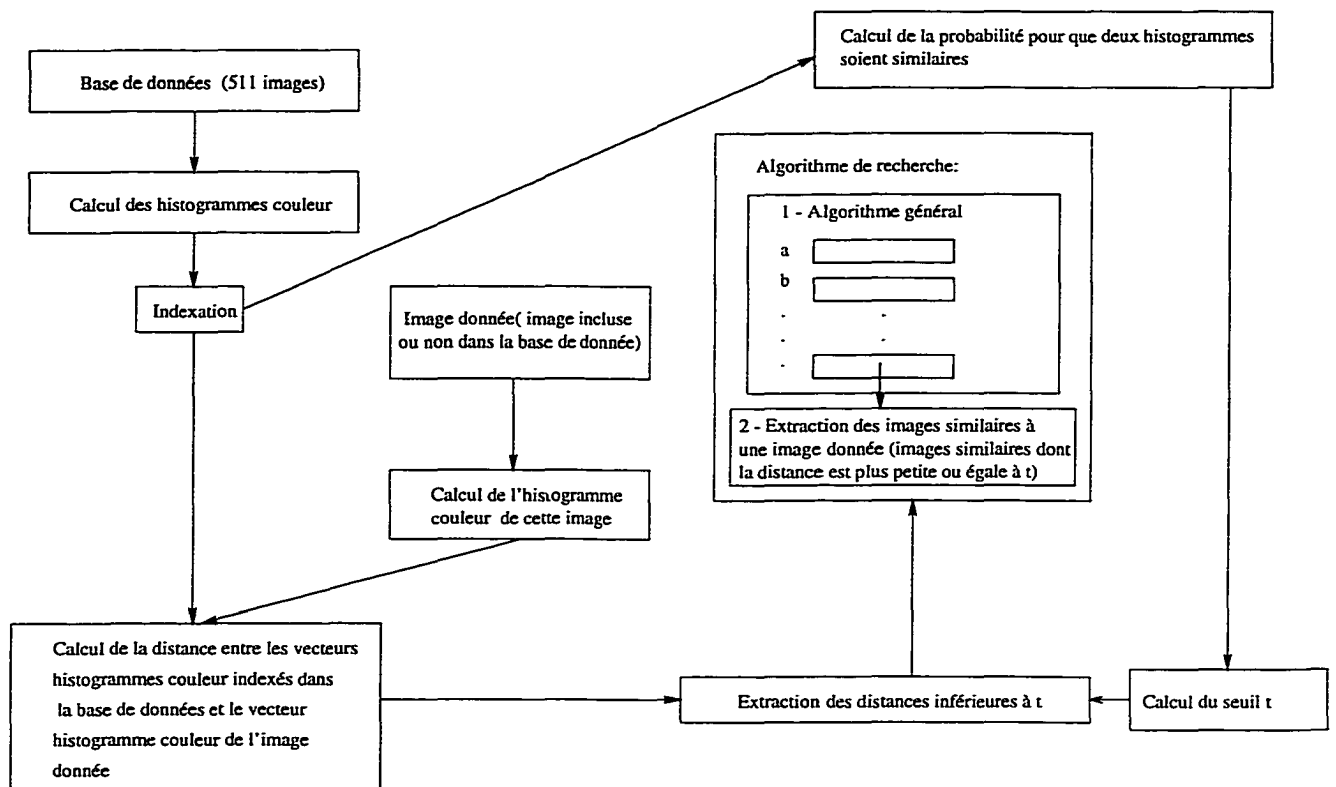


FIG. 1 – Schéma général du système de recherche.

Chaque histogramme admet un indice. Les indices des histogrammes désignent les numéros des images. Soit I_j une image quelconque dans la base de données. Pour connaître les informations relatives à I_j nous trouvons son indice j et son histogramme indexé.

Les histogrammes $h_1, h_2, h_3, \dots, h_{511}$ constituent les enregistrements et leurs indices désignent les clés. Supposons que l'on veuille connaître les informations relatives à l'image dont l'histogramme est h_{101} : le système repère alors le numéro d'enregistrement correspondant. Si c'est le numéro 101, il va alors lire le 101ème enregistrement pour trouver les informations désirées.

Pour extraire les images similaires à une image donnée I , nous calculons les distances entre son histogramme H et tous les histogrammes représentant toutes les images dans la base de données. Nous extrayons les images dont la distance par rapport à I est inférieure ou égale à un certain seuil calculé t . L'utilisateur peut extraire un nombre précis d'images similaires. Dans la section suivante, nous expliquerons la représentation par histogramme et le calcul de la distance entre deux histogrammes quelconques dans la base de données.

1.2 Histogramme couleur et calcul de la distance

1.2.1 Histogramme couleur

Supposons que nous ayons une image M contenant n niveaux de gris. L'histogramme associé à l'image est défini par :

$$H(M) = (h_1, h_2, \dots, h_i, \dots, h_n).$$

h_i est le nombre de pixels ayant le niveau de gris i dans l'image M avec $\sum_{i=1}^n h_i = N$, où N est le nombre de pixels dans l'image M .

Soit I une image couleur, son histogramme H est défini par :

$$H(M) = (H_R, H_G, H_B),$$

où $H_R = (r_1, r_2, \dots, r_j, \dots, r_n)$, $H_G = (g_1, g_2, \dots, g_i, \dots, g_n)$ et $H_B = (b_1, b_2, \dots, b_k, \dots, b_n)$.

Ici r_j est le nombre de pixels de niveau de gris j dans l'image rouge M_R , g_i est le nombre de pixels de niveau de gris i dans l'image verte M_G et b_k est le nombre de pixels de niveau de gris dans l'image bleue M_B . Les vecteurs histogrammes doivent satisfaire la condition suivante :

$$\sum_{j=1}^n r_j = \sum_{i=1}^n g_i = \sum_{k=1}^n b_k = N.$$

Les images de la base ne sont pas de même dimension. Pour que tous les histogrammes des images contiennent le même nombre de pixels, nous effectuons la normalisation. Nous remplaçons $H(M)$ par :

$$H1(M) = \frac{H(M) \times N_0}{N}$$

avec N_0 , facteur de normalisation. Dans notre travail nous avons utilisé $N_0=256$. Dans la figure 2 nous avons une image et son histogramme normalisé. Dans les sections suivantes, nous utilisons H (ou $H(M)$) pour désigner l'histogramme normalisé.

1.2.2 Calcul de la distance

Soit E un espace métrique. Nous définissons la distance d de $E \times E$ dans R^+ , qui, au couple (x, y) de $E \times E$, fait correspondre un nombre $d(x, y) \geq 0$, appelé distance entre x et y , x, y étant deux points dans l'espace E .

Cette distance doit posséder les trois propriétés suivantes :

- Positivité : $d(x, y) > 0$ si $x \neq y$, et $d(x, x) = 0$.
- Symétrie : $d(x, y) = d(y, x)$
- Inégalité triangulaire : $d(x, z) \leq d(x, y) + d(y, z)$

Soient H_1 et H_2 deux histogrammes qui représentent deux images couleur. Pour calculer la distance entre H_1 et H_2 utilisons la distance Euclidienne définie par :

$$d(H_1, H_2) = \sqrt{\sum_{j=1}^n (r_{1j} - r_{2j})^2 + \sum_{j=1}^n (g_{1j} - g_{2j})^2 + \sum_{j=1}^n (b_{1j} - b_{2j})^2} \quad (1.1)$$

avec, $H_1 = (r_{11}, r_{12}, \dots, r_{1j}, \dots, r_{1n}, g_{11}, g_{12}, \dots, g_{1j}, \dots, g_{1n}, b_{11}, b_{12}, \dots, b_{1j}, \dots, b_{1n})$ et $H_2 = (r_{21}, r_{22}, \dots, r_{2j}, \dots, r_{2n}, g_{21}, g_{22}, \dots, g_{2j}, \dots, g_{2n}, b_{21}, b_{22}, \dots, b_{2j}, \dots, b_{2n})$. Nous pouvons aussi utiliser une autre distance comme celle de Manhattan définie par :

$$d1(H_1, H_2) = \sum_{j=1}^n |r_{1j} - r_{2j}| + \sum_{j=1}^n |g_{1j} - g_{2j}| + \sum_{j=1}^n |b_{1j} - b_{2j}| \quad (1.2)$$

Plusieurs chercheurs ont utilisé ces deux distances pour implanter leurs algorithmes. Par exemple : Equitz et al. (1993) [4] ; Stricker, (1992) [5]. Dans notre travail nous utilisons la distance Euclidienne.

Deux histogrammes sont similaires si leur distance est inférieure ou égale à un seuil fourni t . Dans le cas contraire, nous dirons qu'ils sont différents. La notion de similarité joue un rôle important dans l'analyse des propriétés métriques de l'espace d'histogrammes.

1.2.3 Espace d'histogrammes

Dans cette section nous expliquons la définition de l'espace d'histogrammes que nous allons utiliser dans d'autres sections.

L'espace d'histogrammes est un sous-ensemble des vecteurs de dimension $3n$. Soit \mathcal{H} cet espace, \mathcal{H} est défini par :

$$\mathcal{H} = \{(h_1, \dots, h_{3n}) | h_i \geq 0, 1 \leq i \leq 3n, \sum_{i=1}^n h_i = \sum_{i=n+1}^{2n} h_i = \sum_{i=2n+1}^{3n} h_i = N\} \quad (1.3)$$

\mathcal{H} peut être décomposé sous la forme :

$$\mathcal{H} = (\mathcal{H}_r, \mathcal{H}_g, \mathcal{H}_b),$$

où \mathcal{H}_r , \mathcal{H}_g et \mathcal{H}_b sont des sous ensembles de l'espace de vecteurs des dimensions n de la forme :

$$\{(h_1, \dots, h_n) | h_i \geq 0, 1 \leq i \leq n, \sum_{i=1}^n h_i = N\}$$

La section suivante présente la distribution des distances entre les histogrammes et la section 1.2.5 étudie la capacité de l'espace d'histogrammes.

1.2.4 Distribution des distances

Avant d'expliquer les raisons de l'étude de la distribution des distances, il faut définir le nombre de paires d'histogrammes correspondant à une distance donnée t . Soit np ce nombre, np est le nombre de paires d'images dans la base de données dont la distance est plus petite ou égale à une valeur donnée t .

Pour fixer l'intervalle des valeurs raisonnables de t , nous étudions la distribution des distances entre les histogrammes de notre base de données (figure 3). La figure 3 donne le nombre de paires d'histogrammes np à chaque t donnée, prenons par exemple 2600 paires d'histogrammes dont la distance $t=300$. Pour calculer np , nous extrayons les paires des images dont la distance est plus petite ou égale à t . Si t diminue, l'algorithme de recherche donne un nombre réduit des paires d'histogrammes et si t augmente, l'algorithme de recherche fait augmenter ce nombre. Le nombre de paires d'histogrammes augmente avec t jusqu'à une valeur maximale à partir de laquelle le nombre de paires d'histogrammes commence à diminuer (voir figure 3). Donc nous pouvons prendre l'intervalle des valeurs raisonnables de t comme l'intervalle commençant par la valeur où le nombre paires d'histogrammes est minimal et termine par la valeur où ce nombre est maximal. Dans ce premier intervalle, le nombre de paires d'images similaires croît rapidement. Dans la

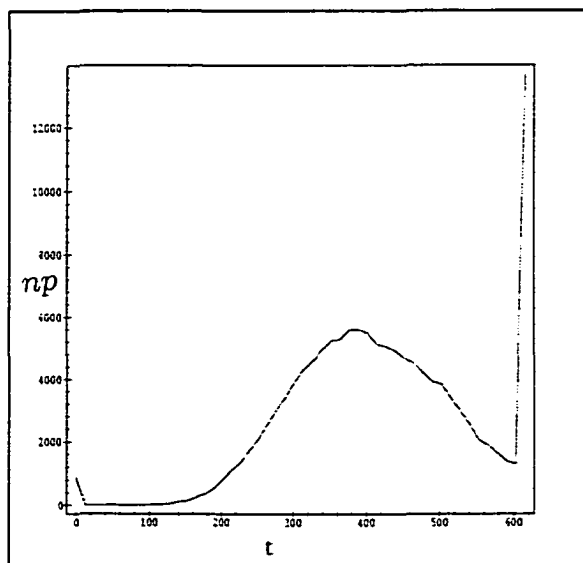


FIG. 3 – *Distribution des distances.*

section suivante nous étudierons la capacité de l'espace d'histogrammes associé à un seuil t donné.

1.2.5 Capacité de l'espace d'histogrammes

Dans cette section nous allons expliquer la définition de la capacité et son utilisation.

Avant de construire une base de données indexée, il faut déterminer si les histogrammes peuvent distinguer les différentes images prises du même environnement. La capacité de l'espace d'histogrammes résout ce problème. Nous examinons le nombre maximal des différents modèles pouvant être stockés dans la base de données. C'est le nombre maximal d'histogrammes "t-différents" dans cette base. Deux histogrammes sont "t-similaires" si leur distance est inférieure ou égale à t . On dit que deux histogrammes sont "t-différents" si leur distance est plus grande que la valeur t donnée.

Soient \mathcal{H} , l'espace d'histogrammes défini dans la section 1.2.3, d , une distance métrique

dans l'espace \mathcal{H} et t une distance donnée. La capacité de \mathcal{H} , notée par $C(\mathcal{H}, d, t)$, est définie comme le nombre maximal d'histogrammes "t-différents" dans l'espace \mathcal{H} .

La valeur de la capacité est maximale dans le cas où tous les histogrammes sont "t-différents". Le calcul de la capacité est difficile si la base de données est très large mais il est possible de l'estimer par des théories de codage (Van Lint, 1992) [15].

Estimation de la capacité

Soit $A(n, 2l, w)$ la fonction qui représente le nombre maximum de différents mots codés en binaire où, n est la longueur de chaque mot, l est la distance de Hamming et w est une constante. $A(n, 2l, w)$ est donnée par la formule :

$$A(n, 2l, w) = \max_{q^{l(w)-1}} \binom{n}{w}, \text{ où} \quad (1.4)$$

$l(w) = \frac{1}{2}(\frac{tw}{N})^2$, q est le plus petit nombre premier suivant n ($q \geq n$) et w est une constante entre 1 et n (Van Lint, 1992) [15].

Supposons que chaque histogramme est représenté par un mot binaire de longueur n et $A(n, 2l, w)$ est le nombre maximum d'histogrammes "t-différents". La capacité satisfait l'inégalité suivante :

$$C(\mathcal{H}, d, t) \geq A(n, 2l, w) \text{ (Markus Stricker, 1994) [2], où}$$

$$w \leq \beta \text{ et } \beta = \min(n, [(\frac{2N}{t})^2]).$$

Donc nous pouvons estimer la capacité par :

$$C(\mathcal{H}, d, t) \approx A(n, 2l, w).$$

Le tableau suivant contient des valeurs calculées de l'estimation de la capacité à un t donné. Nous calculons la valeur de $l(w)$, avec $w=1$ (w est une constante donné), $N=256$

t	$l(w)$	$A(n, 2l, w)$
100	0.08	451.91
150	0.17	217.57
200	0.31	98.01
250	0.48	31.67
300	0.69	7.85

TAB. 1 – Valeurs de la capacité à un t donné.

et t est donné. Nous calculons $A(n, 2l, w)$ et puis nous prenons leur valeur comme estimation de la capacité. Nous remarquons que si t augmente, $l(w)$ augmente et $A(n, 2l, w)$ diminue.

1.3 Les paramètres d'optimisation du temps de recherche

Le but de cette section est de minimiser le nombre d'opérations dans nos algorithmes et de réduire le temps de recherche. Pour atteindre ce but nous calculons la valeur d'un certain t comme seuil. Puis nous extrayons les images dont la distance par rapport à l'image donnée est inférieure ou égale à t . Nous allons expliquer la probabilité de similitude entre deux histogrammes qui nous aide à calculer t . Puis la relation entre le nombre moyen d'images similaires dans la base de données et la probabilité de similitude.

1.3.1 Probabilité de similitude et relation avec le nombre moyen des images similaires

La probabilité de similitude entre deux histogrammes est la probabilité pour que ces deux derniers soient semblables. La probabilité de similitude dépend de la valeur d'une certaine distance donnée t . Cette probabilité est définie par la formule :

$$p(t) = \frac{n_2}{n_1} \quad (1.5)$$

avec, n_1 le nombre total de paires d'histogrammes dans la base de données, $n_2=n_2(t)$ le nombre de paires d'histogrammes "t-similaires". Le tableau suivant contient quelques valeurs calculées de $p(t)$ à chaque t donnée.

Soit R le nombre moyen d'images "t-similaires" dans la base de données. Nous explici-

t	$p(t)$
150	0.05
250	0.1
300	0.3
400	0.4

TAB. 2 – Valeurs de la probabilité à chaque t donnée.

terons, dans le paragraphe suivant comment R peut être utilisé pour calculer la valeur d'un certain seuil dont le rôle est de caractériser "la similarité" acceptable entre l'image requête et les images de la base. Pour cela nous devons estimer R . Selon [2], l'estimation du nombre moyen R est donnée par :

$$R(n, t) \approx n \times p(t), \quad (1.6)$$

où n est le nombre d'images dans la base. La relation établie par la formule (1.6) est

très utile. En effet, il est possible de pré-calculer pour un certain nombre de valeurs de t , la probabilité $p(t)$ (voir section 1.3.3). De même nous pouvons déduire la valeur du seuil t à partir de la relation entre $R(n, t)$ et $p(t)$ (1.6). Dans la section suivante, nous allons expliquer comment trouver la valeur du seuil t .

1.3.2 Calcul du seuil t

L'utilisation du seuil t permet d'éviter de nombreux calculs et de réduire considérablement le temps de recherche. En effet, dans ce cas nous diminuons le nombre d'images à classer (selon la distance) en ne prenant que celles dont la distance à l'image donnée est inférieure ou égale au seuil t . Ceci implique bien sûr moins d'opérations dans nos algorithmes et donc un gain de temps.

La formule (1.6) nous permet de déduire une valeur de seuil appropriée. Nous calculons $p(t)$ par cette formule en donnant à $R(n, t)$ une valeur précise. Nous comparons $p(t)$ avec des valeurs de la probabilité $p(t_i)$ pré-calculées à chaque t_i donné (tableau 2). Puis nous retiendrons du tableau de $p(t_i)$ la valeur de t_i dont la probabilité donne la meilleure approximation de $p(t)$.

1.3.3 Estimation de la probabilité

Dans plusieurs applications, nous ne pouvons prendre la base de données complètement. Dans le cas d'une base de données très grande, il est fastidieux de calculer ou de recalculer à l'aide de la formule 1.5 la probabilité quand une nouvelle image est ajoutée à cette base. Pour ces deux raisons, il faut estimer la probabilité 1.5. Pour faire l'estimation, supposons que les histogrammes sont distribués uniformément dans des sous groupes de

leur espace. L'estimation de la probabilité est donnée par la formule :

$$p_1(t) \approx X(t, NbImages, m) = \frac{1}{m * NbImages} \sum_{j=1}^m B(I_j, t, NbImages) \quad (1.7)$$

avec $\{I_j | j = 1, \dots, m\}$, un ensemble d'images dans la base de données. m , le nombre d'images de cet ensemble. $B(I_j, t, NbImages)$, le nombre de toutes les images dont la distance à I_j est inférieure ou égale à un certain t , où t est le rayon d'une sphère centrée en I_j . Nous supposons que chaque image de l'ensemble est un centre d'une sphère dont le rayon égal à t . $NbImages$ est le nombre total d'images dans la base de données. $\frac{1}{NbImages} B(I_j, t, NbImages)$ peut être interprété comme l'estimation de la probabilité $p(t)$ pour que deux histogrammes soient similaires (1.5). $X(t, NbImages, m)$ est la moyenne de l'estimation de la probabilité. Nous prenons un échantillon de m histogrammes loin les uns des autres. C'est-à-dire que la distance entre deux histogrammes quelconques dans l'échantillon doit être supérieure ou égale à t . Dans le cas contraire où les histogrammes de l'échantillon sont proches, l'estimation n'est pas bonne parce que les histogrammes sont mal distribués dans leur espace. D'après l'expérience nous trouvons que $t = \frac{1}{m}\%$ de la valeur de la distance entre les deux images les plus éloignées dans la base de données. La probabilité et son estimation calculées sont représentées par la figure 4. La première courbe représente la fonction de la probabilité $p(t)$ et la deuxième représente son estimation $p_1(t)$ à chaque t donné. $p(t)$ et $p_1(t)$ croient avec la valeur de t et à chaque t correspond une valeur de la probabilité (formule 1.5) et de son estimation (formule 1.7). Pour savoir si l'estimation est bonne ou non, nous faisons une comparaison entre la probabilité et son estimation. Si leurs valeurs sont proches nous pouvons dire que l'estimation est bonne.

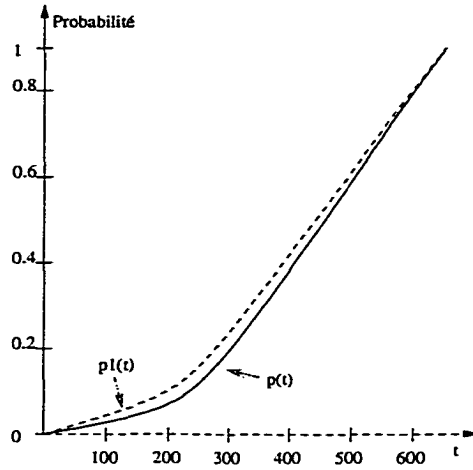


FIG. 4 – *La probabilité et son estimation.*

1.4 Algorithmes

Dans cette section, nous présentons les algorithmes que nous avons utilisés pour indexer la base de données et pour rechercher les images similaires à une image donnée par l'utilisateur.

1.4.1 Algorithme 1 : Indexation de la base de données

Nous nous appliquerons ici à décrire l'algorithme d'indexation. Transformons l'image de format JPEG en une image plus simple à utiliser. A cet effet, nous prendrons le format PPM. Coupons ensuite l'en-tête de l'image format PPM. Ensuite calculons et normalisons l'histogramme [1] parce que les images ne contiennent généralement pas le même nombre de pixels. Chaque histogramme est représenté par un vecteur. Nous le mettons dans un index de la base de données [2].

Algorithme 1 : Pour faire l'indexation de la base de données

```
{  
  lire  $NbImages$  ;  
  pour  $i = 1$  à  $NbImages$  faire {  
    lire image  $i.jpeg$  ;  
    calculer son histogramme  $(H_R, H_G, H_B)$  ;  
    normaliser  $(H_R, H_G, H_B)$  ;  
    mettre  $(H_R, H_G, H_B)$  dans un vecteur  $v_i$  ;  
    indexer  $v_i$  dans un fichier ;  
  }  
}
```

$NbImages$: nombre total d'images dans la base de données.

$i.jpeg$: image quelconque dans la base de données, avec $1 \leq i \leq NbImages$.

1.4.2 Algorithme 2 : Recherche des images similaires à une image donnée

Cet algorithme est conçu pour retrouver les images similaires à une image donnée. Nous lisons et visualisons, puis calculons l'histogramme de cette image. Calculons les distances entre le vecteur "histogramme" de cette image et tous les vecteurs "histogrammes"

stockés dans un index de la base de données [2]. Chaque image est représentée par un histogramme dans l'index. Enfin extrayons les images dont la distance avec l'image donnée est inférieure ou égale à un certain seuil calculé.

Algorithme 2: Pour trouver les images similaires à une image donnée

```

{
lire image donnée;
visualiser cette image;
calculer son histogramme  $H(image) = (H_1, H_2, H_3)$ ;
lire  $R$  le nombre moyen d'images similaires donnée;
calculer du seuil  $t$  correspond à  $R$  (formule 1.6);
trouver les images  $\{I\}$  de la base de données dont la
distance (1.1),  $d(H(image), H(I)) \leq t$ ;
afficher ces images par ordre croissant de distance
 $d(H(image), H(I))$  (formule 1.1);
}

```

1.4.3 Algorithme 3: Calcul du seuil qui contrôle le nombre moyen d'images semblables

Nous utilisons le seuil pour “contrôler” les images similaires désirées. Dans cet algorithme nous déterminons les distances minimale et maximale entre deux images quelconques dans la base de données. La distance minimale est celle qui sépare les deux images les plus proches dans la base de données. De façon similaire, nous définissons la distance maximale comme étant la distance séparant les deux images les plus éloignées. Nous calculons ces deux distances à partir de l'équation 1.1. Nous donnons des valeurs de seuil entre ces valeurs minimum et maximum. Nous calculons à l'aide de l'équation 1.5 la probabilité pour que deux histogrammes soient similaires pour chacun de ces seuils. Le nombre moyen d'images similaires R est donné. Puis nous déterminons la probabilité

Algorithme 3 : Pour calculer le seuil à partir du nombre moyen d'images semblables

```
{
  calculer les distances minimale et maximale
    entre deux images quelconques dans la
    base de données (formule 1.1);
  lire NbSeuil le nombre de seuils données;
  pour  $i = 1$  à NbSeuil faire {
    prendre un seuil  $t_i$  entre les valeurs
      minimum et maximum de la distance;
    calculer la probabilité  $p_i$  pour chaque  $t_i$ 
      (formule 1.5);
  }
  lire  $R$  le nombre moyen d'images similaires donnée;
  calculer la probabilité de similitude entre deux
    histogrammes (formule 1.6);
  pour  $i = 1$  à NbSeuil faire {
    si  $p_i \leq p(t)$  et  $p_{i+1} > p(t)$ 
      le seuil  $t = t_{i+1}$ ;
  }
}
```


de similitude entre deux histogrammes à partir de l'équation 1.6. La probabilité retenue (et le seuil correspondant) sera la plus proche de celle du nombre moyen.

1.4.4 Algorithme 4 : Calcul de la probabilité de similitude entre deux histogrammes

Nous utiliserons cet algorithme pour déterminer la probabilité de similitude entre deux histogrammes (1.5) pour un seuil donné.

Algorithme 4 : Pour calculer la probabilité de similitude entre deux histogrammes

```
{
  lire  $NbSeuil$  le nombre des seuils données ;
  calculer  $n_1$  le nombre total de paires d'histogrammes
    dans la base de données ;
  pour  $i = 1$  à  $NbSeuil$  {
    lire le seuil  $t_i$  ;
    calculer  $n_2$  le nombre de paires d'histogrammes
      " $t_i - similaires$ " ;
    calculer  $p_i(t)$  la probabilité de similitude entre
      deux histogrammes de la base de données
      (formule 1.5) ;
  }
}
```

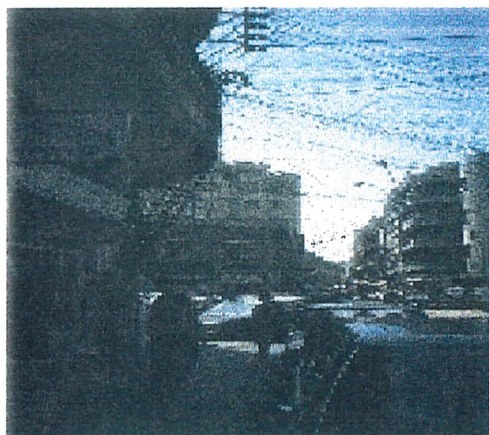
1.5 Résultats

Dans cette section, nous présentons quelques résultats que nous avons obtenus. Nous trouvons, pour chaque exemple, le nombre moyen R d'images similaires, le seuil t et le temps de recherche tr . A la fin de cette section nous allons présenter un tableau qui

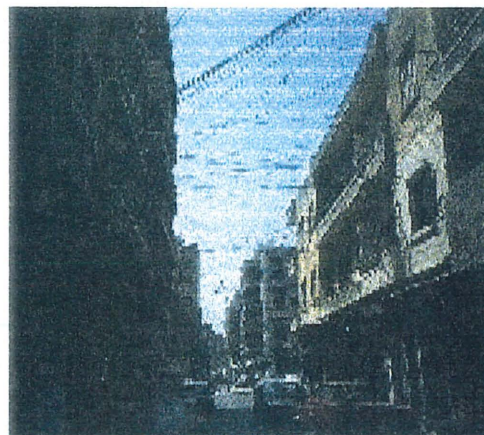
contient : le nombre moyen d'images semblables R , le seuil t calculé à partir de R , la moyenne d'images semblables MIS calculée en parcourant toute la base de données. Dans la figure 5 nous avons une image dans un quartier de Beyrouth. L'image originale est dans (a). Nous avons trouvé deux images similaires à celle-ci. La première image (b) est l'image dont la distance à l'image originale est nulle (l'image (b) est l'image originale elle-même). La deuxième image (c) est prise dans le même quartier. $R = 5$, $t = 170.9$ et $tr = 87s$. Dans cette figure nous remarquons la similarité entre les images (a), (b) et (c). Dans la figure 6 nous avons les histogrammes des images de la figure 5. La distance entre les deux histogrammes (a) et (b) est nulle et la distance entre les deux histogrammes (a) et (c) est petite. La figure 7 montre un autre exemple de recherche des images similaires. L'image originale est (a). En plus de trouver l'image originale, l'algorithme trouve deux autres images. $R = 5$, $t = 170.9$ et $tr = 40s$. L'image (c) ressemble beaucoup à l'image originale (a). Malgré le fait que les histogrammes de l'image originale et l'image (d) se ressemblent, elles sont perçues différemment par l'oeil. Dans la figure 8 nous avons les histogrammes des images de la figure 7. Dans la figure 9, l'image originale (a) est un plan d'un autre quartier de Beyrouth. Nous avons trouvé quatre images similaires à celle-ci. $R = 5$, $t = 170.9$ et $tr = 30s$. Les images (b) et (c) ressemblent beaucoup à ce plan tandis que (d) et (e) ne lui ressemblent pas. La distance entre les histogrammes des images (d), (e) et l'histogramme de l'image originale est inférieure ou égale au seuil calculé mais ces images ne ressemblent pas. Dans la figure 10 l'image originale est (a). Nous avons trouvé cinq images similaires à l'image originale dont trois ((b), (c), (d)) lui ressemblent beaucoup, $R = 5$, $t = 170.9$ et $tr = 44s$. Dans la figure 11 nous avons utilisé l'image (c) de la figure 10 comme image originale. Nous avons obtenu le même résultat que dans l'exemple précédent mais l'ordre des images est changé parce qu'il est relatif au degré de similitude avec l'image, $R = 5$, $t = 170.9$ et $tr = 45s$.



(a)



(b)



(c)

FIG. 5 – (a) Image originale. La première image (b) est l'image dont la distance à l'image originale est nulle. Le nombre moyen d'images $R = 5$, $t = 170.9$ et $tr = 87s$.

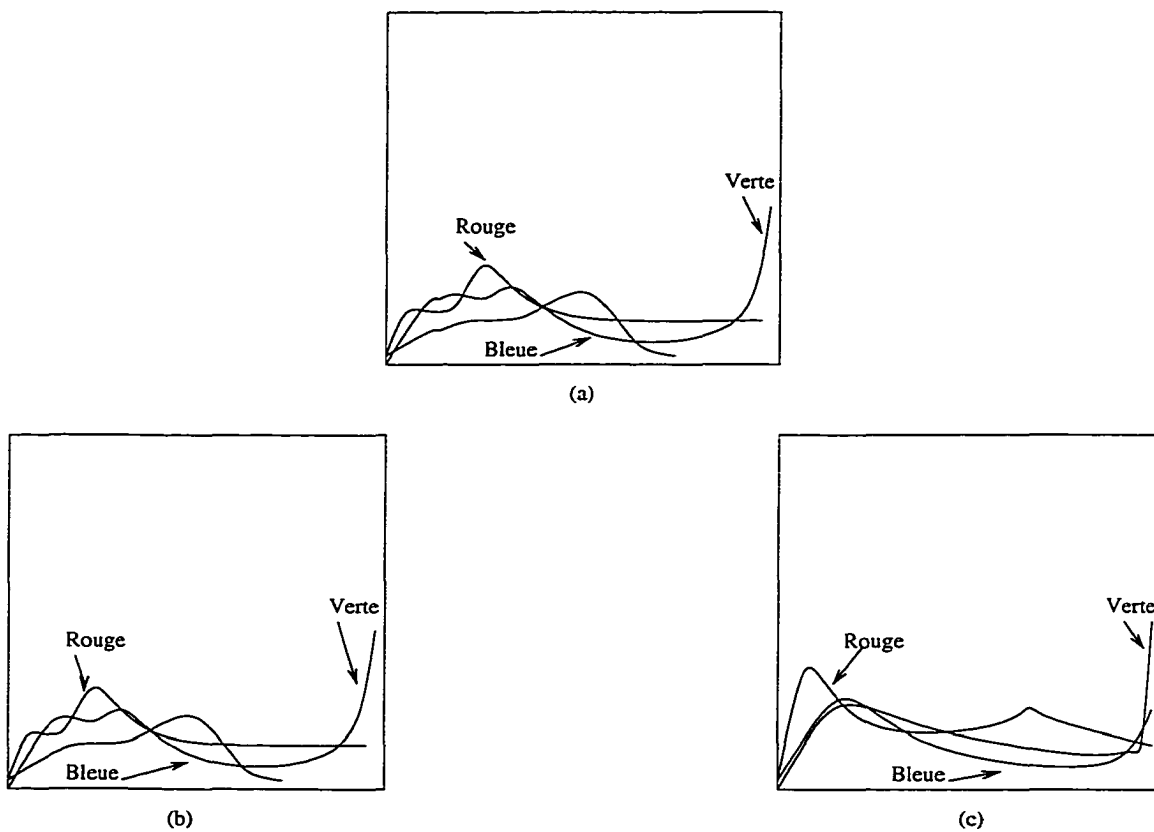


FIG. 6 – La distance entre les histogrammes (a) et (b) est nulle et la distance entre les histogrammes (a) et (c) est petite.

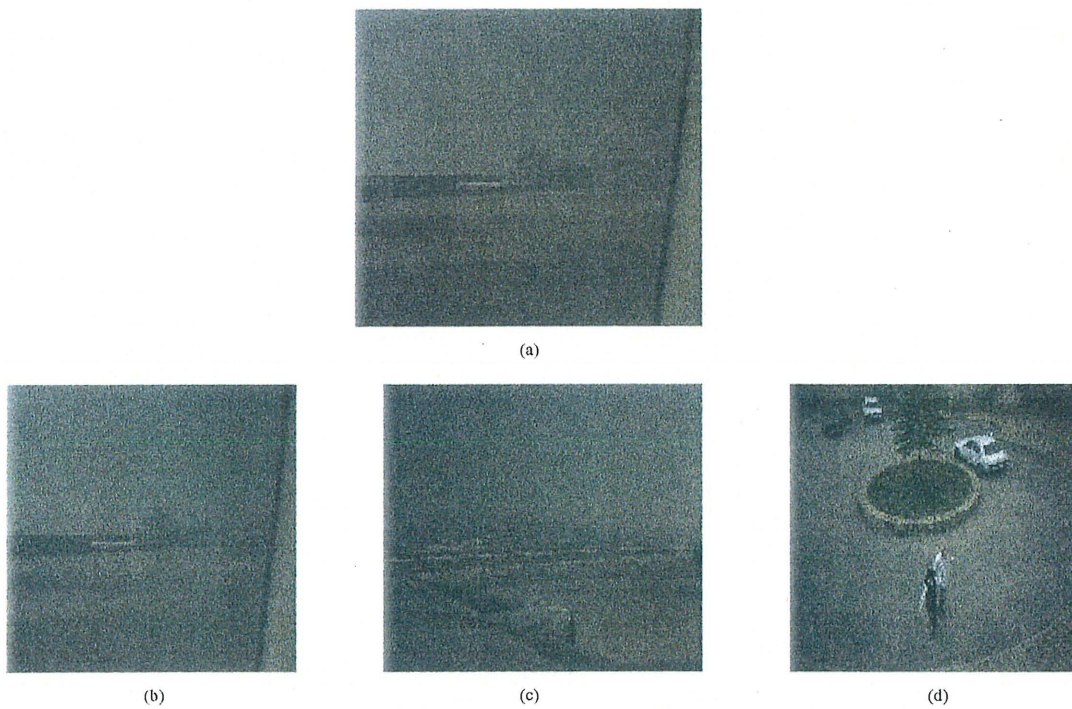


FIG. 7 – *Trois images similaires à l'image originale. Le nombre moyen d'images $R = 5$, $t = 170.9$ et $tr = 40s$. Le seuil t dépend de R .*

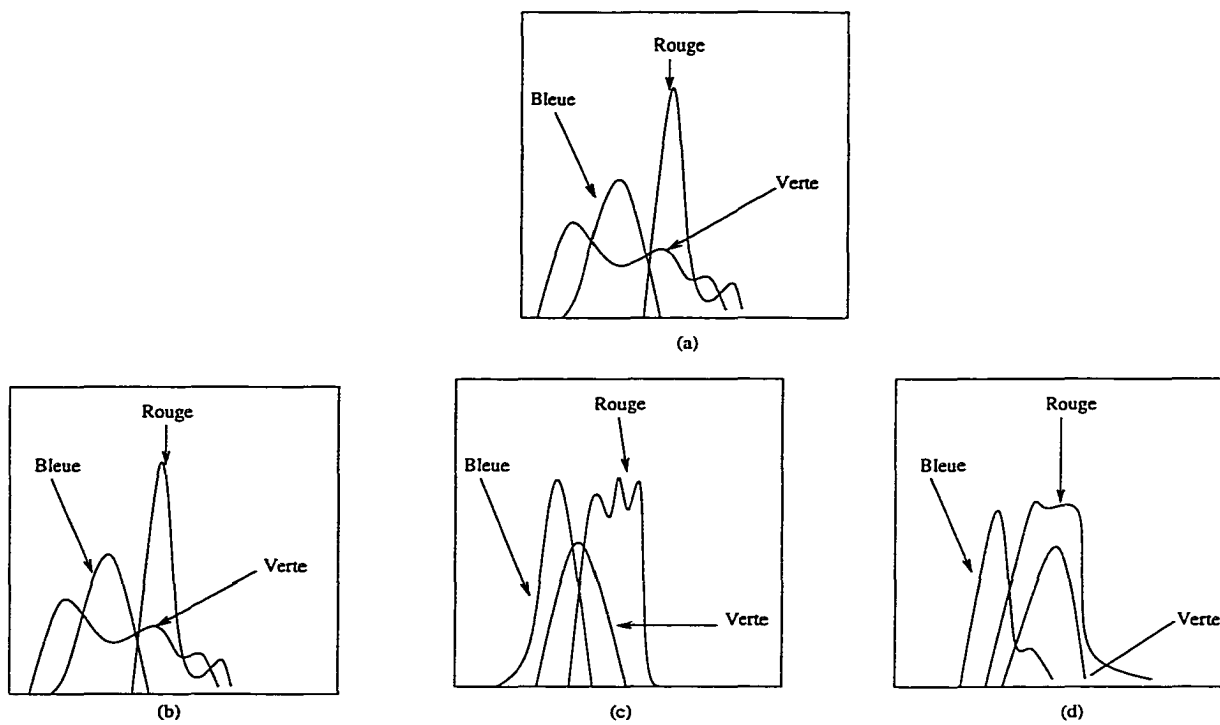


FIG. 8 – (a) Histogramme de l'image originale. (b), (c) et (d) sont les histogrammes des images similaires à l'image originale. Les histogrammes se ressemblent mais les images pas nécessairement.



(a)



(b)



(c)



(d)



(e)

FIG. 9 – *Quatre images similaires à l'image originale. Le nombre moyen d'images $R = 5$, $t = 170.9$ et $tr = 30s$.*



(a)



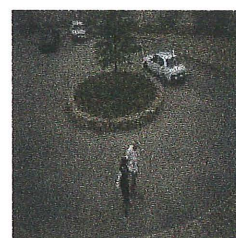
(b)



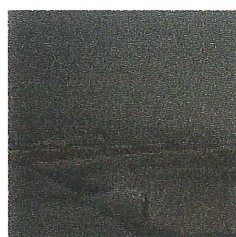
(c)



(d)



(e)



(f)

FIG. 10 – (a) Image originale. Nous avons trouvé cinq images similaires à l'image originale. Le nombre moyen d'images semblables $R = 5$, $t = 170.9$ et $tr = 44s$.

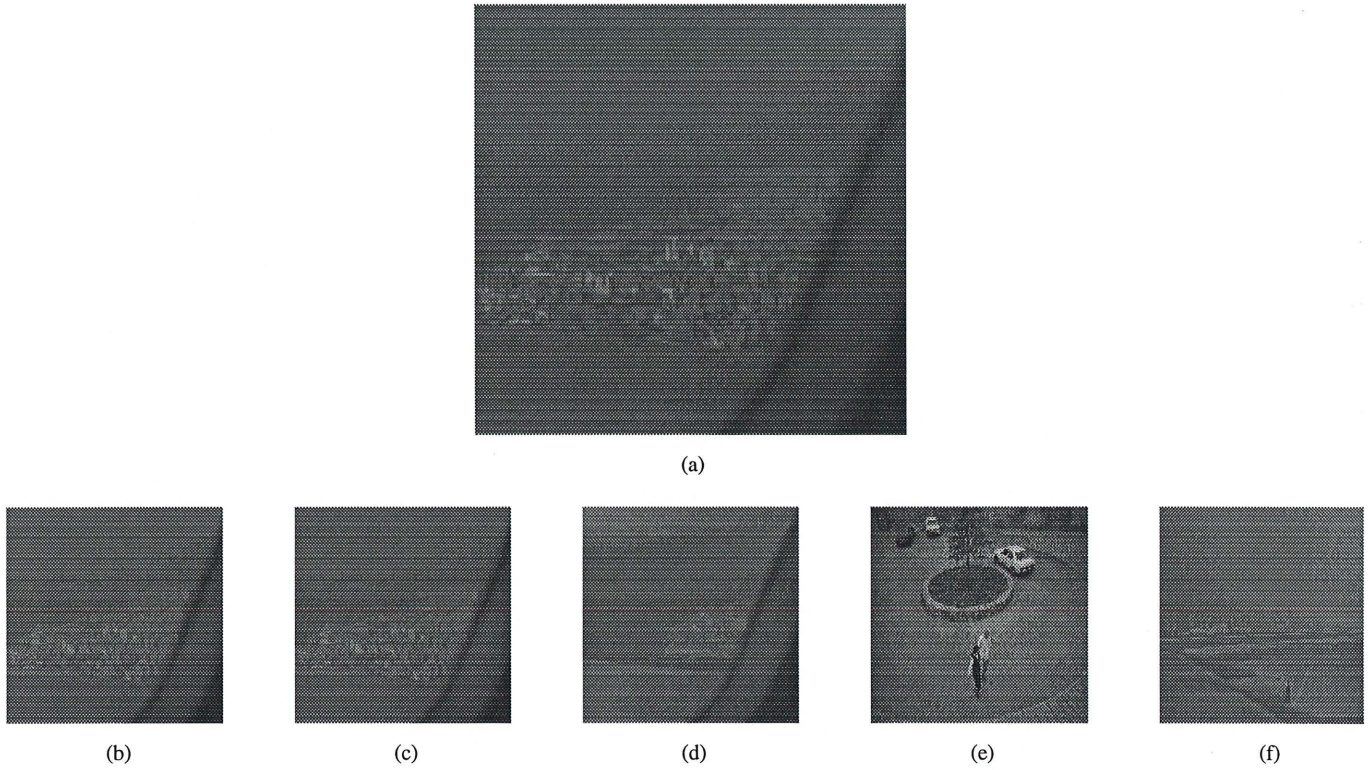


FIG. 11 – L'image originale est dans (a). Nous avons trouvé cinq images similaires à cette image. L'ordre des images similaires est relatif au degré de similitude décroissant avec l'image originale. $R = 5$, $t = 170.9$ et $tr = 45s$.

R	t	MIS
2	170.9	3.53
5	206.6	7.48141
10	221.9	11.0254
15	237.2	15.5832
20	247.4	19.6477
25	257.6	24.1918
30	262.7	26.8513

TAB. 3 – Moyennes des images semblables pour 511 images, pour chaque valeur de R donnée par l'utilisateur.

1.6 Conclusion

La couleur est une caractéristique très importante dans l'image. Chaque pixel dans l'image couleur est représenté par trois composantes (rouge, verte et bleue). L'image couleur représentée par un histogramme de couleurs. Cet histogramme couleur peut être efficacement utilisé pour rechercher les images semblables à une image donnée. Nous faisons simplement la comparaison entre deux images à partir des leurs histogrammes. Les histogrammes contiennent le même nombre de pixels indépendamment de la taille de l'image. L'histogramme reste invariant si nous faisons rotation ou translation de l'image, mais il varie lentement en fonction de l'angle de vision. La couleur n'est pas suffisante parce que nous pouvons trouver des images qui ne sont pas similaires malgré que leurs histogrammes sont semblables.

Nous avons construit une base de données de 511 images. Chaque histogramme contient un nombre total de pixels égal à 3×256 . Pour chaque image donnée nous trouvons au moins une image semblable dont la distance est égale à 0. Nous pouvons trouvé au moins l'image elle-même. Le nombre d'images similaires à une image donnée dépend du seuil calculé à partir du nombre moyen d'images similaires donné par l'utilisateur. Le nombre d'images similaires augmente avec le seuil calculé.

Pour indexer la base de données et construire le système de recherche des images similaires, nous avons créé trois programmes et plusieurs fonctions ayant en tout 1500 lignes codés en C^{++} .

Dans le cas où il y a une base de données très grande, c'est difficile de calculer la distance entre l'image donnée et toutes les images de la base. Dans le chapitre suivant, nous allons présenter une méthode qui compare l'image donnée avec un certain nombre des images

significatives de la base.

Chapitre 2

Indexation à partir des réseaux neuronaux SOFM

Les réseaux de neurones artificiels sont inspirés du fonctionnement du cerveau humain et de son système nerveux. Ils peuvent résoudre des problèmes complexes, difficilement traitables par les techniques courantes comme les modèles mathématiques traditionnels. Les réseaux de neurones artificiels établissent la fonction de correspondance d'un domaine d'entrée vers un domaine de sortie à partir d'un ensemble restreint d'exemples typiques de l'application à traiter, et font des généralisations significatives.

Le but de l'utilisation des réseaux neuronaux dans notre travail est d'éliminer un certain nombre des comparaisons et d'accélérer notre système de recherche des images semblables à une image donnée. En effet la méthode que nous avons expliquée dans le chapitre précédent, fait une comparaison de l'image donnée avec toutes les images de la base. Dans la méthode que nous allons expliquer, nous faisons une nouvelle indexation de la base d'images nous permettant de comparer l'image donnée avec une partie des images de la base. De cette façon, nous éliminons un grand nombre de comparaisons et le temps de recherche sera moins long. Pour réduire le nombre des comparaisons et

accélérer notre système de détection des images, il faut rechercher une méthode qui pouvant réorganiser la base d'images d'une façon efficace. Pour faire cela, nous avons utilisé les réseaux SOFM(Self-Organizing Feature-Mapping) pour calculer plusieurs centres afin de les comparer avec l'image donnée. nous trouvons le centre le plus proche à l'image donnée, c'est à dire le centre dont la distance par rapport à cette image est minimum. Enfin, nous comparons cette image avec l'ensemble des images liées à ce centre et à ceux voisins de celui-ci.

La hiérarchie (figure 12) décrit les étapes nécessaires pour faire la nouvelle indexation de la base d'images et pour faire sortir les images semblables à l'image donnée en utilisant la méthode de SOFM. Dans cette recherche nous comparons l'image donnée avec les centres calculés. Pour faire cette comparaison, nous calculons l'histogramme de l'image donnée ainsi que les distances entre cet histogramme et les histogrammes représentant les centres. Nous cherchons le centre le plus proche de cette image, c.à.d que la distance le séparant de l'histogramme de cette image, est minimum. Puis nous comparons l'image donnée avec le nuage de ce centre. Le nuage d'un centre est l'ensemble des images de la base de données dont la distance par rapport à ce centre est inférieure ou égale à un certain seuil calculé. Enfin nous extrayons les images dont la distance avec l'image donnée est plus petite ou égale à un certain seuil calculé. Dans ce chapitre, nous allons expliquer ces étapes et les résultats obtenus. Dans la section suivante, nous allons présenter deux points, le fonctionnement des réseaux neuronaux et l'apprentissage à partir des ces réseaux. Ces deux points sont nécessaires pour comprendre la méthode SOFM. Dans la section 2.2 nous allons expliquer cette méthode pour calculer les centres afin de réorganiser notre base de données. Nous allons présenter les avantages de cet algorithme et démontrer son efficacité dans notre travail. La section 2.3 est pour expliquer la nouvelle indexation de la base d'images et extraire les images semblables à l'image donnée à partir de cette indexation. Dans la section 2.4 nous allons présenter quelques résultats obtenus à partir des deux méthodes, parcours général de la base d'images, parcours des centres

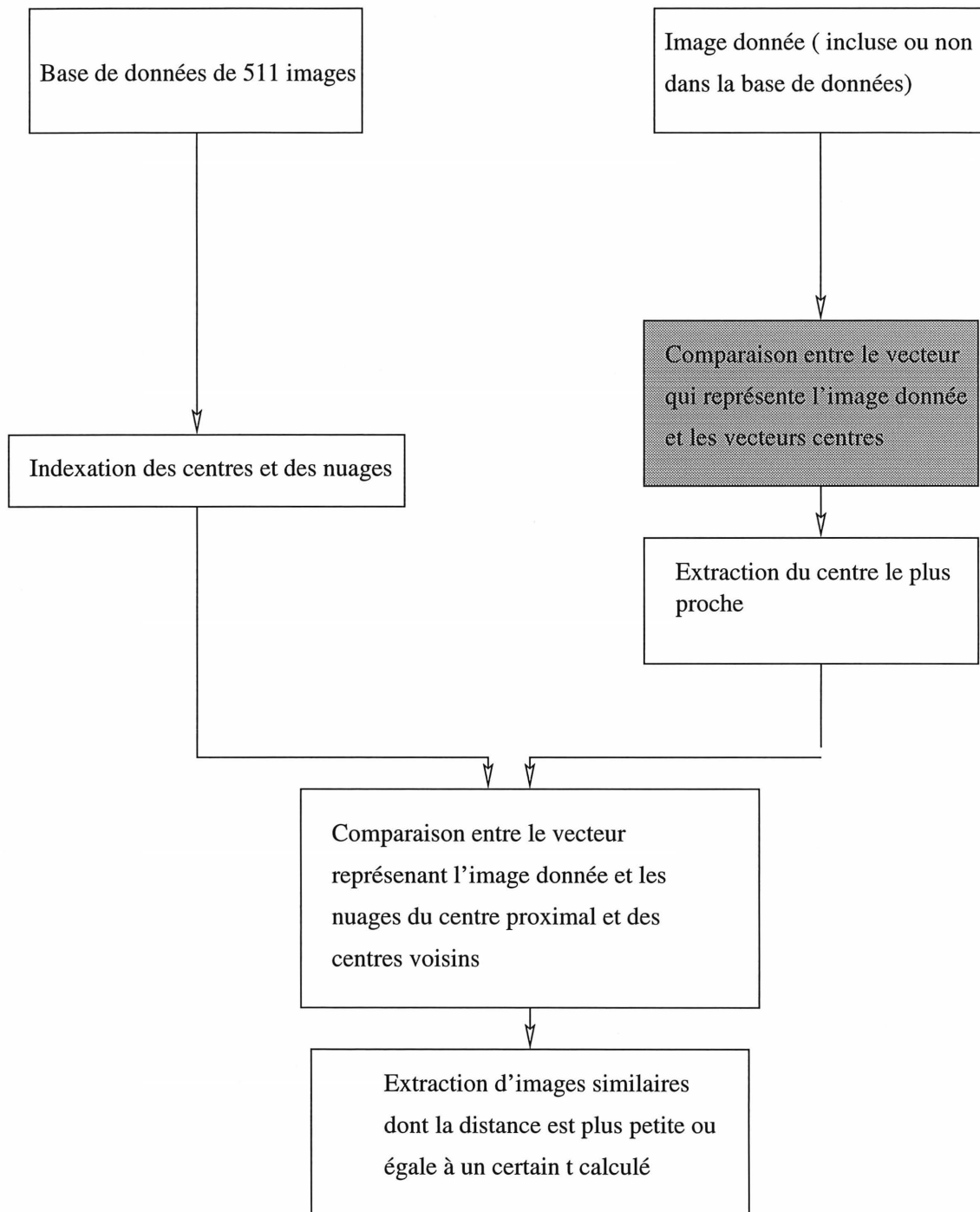


FIG. 12 – *Système de recherche à partir des centres.*

et temps de recherche dans chaque méthode. Ces résultats démontrent le fonctionnement et l'efficacité de SOFM par rapport à l'autre méthode.

2.1 Fonctionnement et apprentissage

Dans cette section nous allons expliquer le fonctionnement des réseaux neuronaux et les différentes sortes d'apprentissage à partir de ces réseaux. Cette section est comme une introduction pour bien comprendre le fonctionnement de la méthode SOFM qui est une sorte d'apprentissage non supervisé.

Généralement, un réseau de neurones artificiels (RNA) reçoit des données à l'entrée, les traite, et produit des données de sortie. Chaque couche de neurones accède et traite les données de façon simultanée. Le processus de propagation des activations entre les couches peut se faire de différentes façons. Parmi les plus courantes, on retrouve la propagation en avant du signal de la couche d'entrée vers la couche de sortie. Un RNA de trois couches exigera alors trois cycles d'opération, un pour l'entrée des données par la couche d'entrée, un deuxième pour le traitement par la couche intermédiaire et finalement un troisième où le RNA fournit la sortie de traitement. Il existe aussi la propagation récurrente, il s'agit souvent de réseaux multicouches dans lesquels les sorties sont redirigées vers l'entrée avec un certain délai.

L'apprentissage permet de modifier les poids et ainsi de fixer la représentation interne que le RNA doit avoir pour résoudre un problème quelconque. Cet apprentissage s'effectue en modifiant les poids des connexions avec l'aide de données d'apprentissage tirées du problème à traiter. Dans son apprentissage, le RNA doit être capable de former des généralisations significatives à partir d'un ensemble d'apprentissage restreint. L'apprentissage est généralement divisé en deux catégories, soit l'apprentissage supervisé et l'apprentissage non supervisé. Dans l'apprentissage supervisé, nous fournissons les données d'entrée et les données de sortie désirée pour l'application en question, et l'apprentissage tente

de reproduire cette correspondance entre le domaine d'entrée et le domaine de sortie. L'apprentissage est souvent contrôlé par l'erreur faite par le RNA en cours d'apprentissage. L'apprentissage non supervisé ne requiert pas une représentation directe des sorties désirées lors de l'apprentissage. L'apprentissage non supervisé (auto-organisé) vise à la découverte de certaines caractéristiques, souvent statistiques, des données. Chacune de ces lois d'apprentissage présente des particularités pouvant favoriser certains modèles des réseaux neuronaux. Après cette introduction, nous allons présenter dans la section suivante la méthode SOFM que nous avons utilisée dans notre travail.

2.2 Réseaux SOFM

Le but de l'utilisation de SOFM est de restructurer la base d'images pour nous permettre de comparer l'image donnée avec un certain nombre d'images et non pas avec toutes les images de cette base. Dans cette section, nous allons expliquer cette méthode et son utilisation mais avant de commencer, il faut passer brièvement par la méthode de k-moyenne qui est une forme simplifiée de SOFM.

La méthode de k-moyenne est une méthode non supervisée, utilisée pour déterminer les centres des groupes de données. L'algorithme de k-moyenne se résume comme suit :

- 1- Initialiser le nombre des centres k .
- 2- Initialiser les centres des groupes $\mu_j \mid j = 1, \dots, k$.
- 3- Trouver le centre μ_j le plus proche à $x(n)$: Étant donné un vecteur d'entrée $x(n)$, trouver j_0 tel que :

$$\|x(n) - \mu_{j_0}\|^2 = \min \|x(n) - \mu_j\|^2 \mid j = 1, \dots, k$$

- 4- Mettre à jour μ_j par :

$$\delta\mu_j = \eta(x(n) - \mu_j), \text{ où } \eta \text{ est le taux d'apprentissage.}$$

L'algorithme de SOFM(Self-Organizing Feature-Mapping) a été développé par T.Kohonen [12] pour transformer les signaux (des grandes dimensions) en points sur une carte à une

ou deux dimensions tout en préservant l'ordre topologique. Cette carte est composée de deux couches dont la première constitue l'entrée du réseau, et la seconde (la carte même) la sortie. Les neurones de cette carte sont disposées en une topologie déterminée. Comme pour les autres réseaux, les liens sont ordonnés de façon à faciliter leur accès, c'est-à-dire en regroupant ensemble les liens qui convergent vers un même neurone de sortie.

L'algorithme SOFM peut calculer un certain nombre des centres significatifs dans l'espace d'histogrammes. Les histogrammes des images de la base sont regroupés dans des groupes distincts. A l'intérieur de chacun de ces groupes, un histogramme représente l'ensemble de tous les histogrammes de ce groupe. C'est cet histogramme que nous appelons le centre significatif. A partir de ces centres et de leurs images nous pouvons extraire les images semblables à l'image donnée.

Cet algorithme est résumé comme suit :

- 1- Initialisation des poids w_i .
- 2- Tirer au hasard un échantillon x à partir des données d'entrées.
- 3- Evaluation de la similarité et repérage du neurone "gagnant" :

$$i(x) = \operatorname{argmin}_j \| x(n) - w_j(n) \|$$
- 4 Mise-à-jour des poids

$$w_{k.}(n+1) = \begin{cases} w_{k.}(n) + \eta(n)[x(n) - w_{k.}(n)] & \text{si } k \in \Lambda_{i(x)}(n) \\ w_{k.}(n) & \text{sinon} \end{cases}$$

- 5- Si les changements sont encore significatifs par rapport à l'itération précédente, passer à l'étape 2.

$\Lambda_{i(x)}(n)$, est une fonction de voisinage autour du neurone "gagnant" $i(x)$. Le taux d'apprentissage $\eta(n)$ et le voisinage $\Lambda_{i(x)}(n)$ sont ajustés au cours de l'apprentissage.

Dans notre travail nous comparons l'image donnée avec le nuage du centre le plus proche d'elle, mais il est possible qu'il y ait d'autres images semblables à cette image dans les nuages des centres voisins à ce centre. Là intervient la notion de voisinage présente

dans l'algorithme. Par exemple, nous définissons une structure de voisinages en carré de 8 points. Donc l'apprentissage concurrentiel sera appliquée non seulement au centre mais aussi à ses voisins. Pour assurer la stabilisation de cet apprentissage, différents schémas de décroissance au cours du temps t ont été proposés pour $\eta(n)$ [13]. Sur le plan pratique, une décroissance linéaire en t avec un arrêt à 0 a donné de bons résultats, et une diminution de la taille du voisinage au cours du temps [14] a été proposée pour augmenter les performances de l'apprentissage.

Des études théoriques et expérimentales montrent que l'algorithme SOFM possède de bonnes propriétés. Parmi ces propriétés [13] nous avons :

- Il fournit une bonne approximation de l'espace d'entrée,
- il préserve l'ordre topologique de la carte calculée,
- il reflète les variations dans les statistiques de la distribution d'entrée.

On peut réorganiser notre base de données en utilisant une autre méthode comme la k-moyenne, mais nous trouvons que SOFM est plus efficace et répond bien à nos besoins. SOFM est plus avantageux car il utilise la notion de voisinage, et permet aussi de trouver des images semblables à l'image donnée.

2.3 Nouvelle indexation de la base d'images

Le but de cette section est d'expliquer comment faire la nouvelle indexation de la base d'images et comment rechercher les images semblables à l'image donnée à partir de cette indexation. Dans la méthode expliquée dans le chapitre précédent, chaque histogramme admet un indice. Les indices des histogrammes désignent les numéros des images. Pour extraire les images semblables à l'image donnée il faut parcourir séquentiellement les histogrammes des images de la base à partir de leurs indices et calculer leurs distances avec l'histogramme de cette image. Puis nous extrayons les images dont la distance par rapport à cette image est inférieure ou égale à un certain t calculé. Dans la nouvelle

indexation de la base nous supposons que les histogrammes des images sont regroupés, dans des groupes distincts, dans leur espace et que chaque groupe admet un centre qui le représente. Le nombre des ces centres est déterminé à partir des plusieurs tests. Nous avons essayé l'algorithme SOFM sur plusieurs nombres des centres jusqu'à ce que nous arrivions à un nombre des centres significatifs qui donnent de bons résultats des images semblables et éliminent le nombre d'opérations. La hiérarchie (figure 12) représente les étapes principales pour faire la nouvelle indexation et extraire les images semblables à l'image donnée.

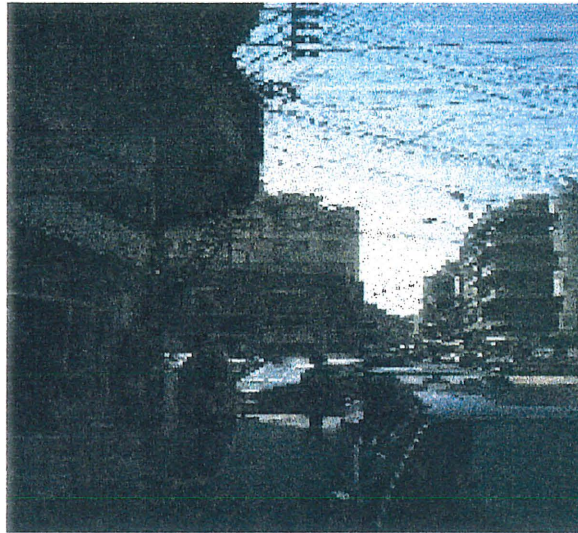
Pour trouver les images semblables à une image donnée, nous comparons cette image avec les centres puis avec le nuage du centre le plus proche. La démarche générale pour la recherche des images semblables à une image donnée, en utilisant la nouvelle indexation de la base est résumée comme suit :

- Comparer l'histogramme de l'image donnée avec tous les centres calculés. Pour faire cette comparaison, calculer les distances entre cet histogramme et les histogrammes représentant les centres.
- Chercher le centre le plus proche de cette image, c.à.d que la distance le séparant de l'histogramme de cette image, est minimum, soit C ce centre.
- Comparer l'image donnée avec le nuage du centre C , c.à.d avec les images distantes de ce centre d'une valeur inférieure ou égale à un certain rayon calculé. Puis comparer l'image donnée avec les nuages des centres voisins de C
- Extraire les images dont la distance à l'image donnée est plus petite ou égale à un certain t calculé.

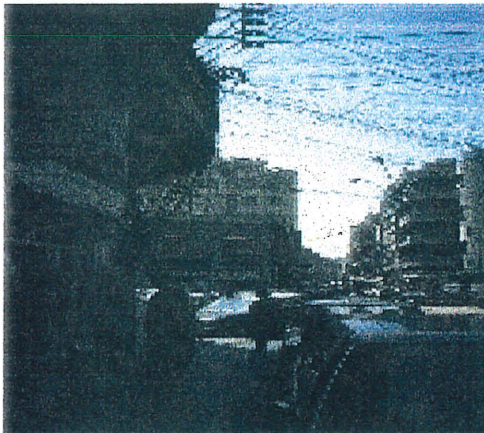
Dans la section suivante, nous allons démontrer l'efficacité de SOFM par rapport à la méthode présentée dans le chapitre précédent. Nous allons présenter quelques résultats obtenus à partir de ces deux méthodes et le temps de recherche dans chaque méthode.

2.4 Résultats

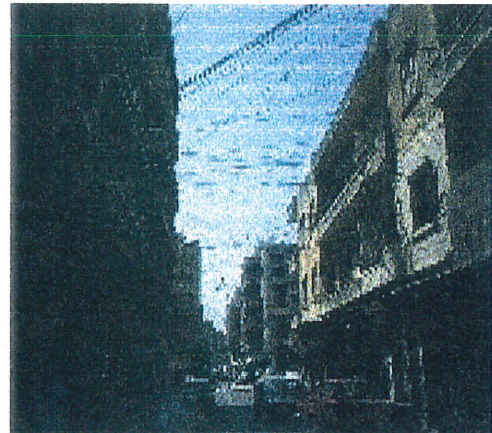
Dans cette section, nous allons présenter quelques résultats intéressants que nous avons obtenus à partir des deux méthodes (parcours général de la base d'images et parcours à partir des centres) et le temps de recherche dans chaque méthode. Nous supposons que t_1 est le temps de recherche par la méthode du parcours général et t_2 , le temps de recherche par la méthode du parcours des centres. Dans la figure 13, nous avons une image (a) dans un quartier de Beyrouth. Nous avons trouvé deux images similaires à celle-ci. La première image (b) est l'image dont la distance à l'image originale est nulle (l'image (b) est l'image originale elle-même), $t_1 = 87s$, $t_2 = 67s$. La deuxième image (c) est prise dans le même quartier. Le nombre d'images moyen donné par l'utilisateur est $R = 5$. Le seuil calculé est de 170.9, $t_1 = 30s$, $t_2 = 11s$. Dans la figure 14, l'image originale (a) est un plan d'un autre quartier de Beyrouth. Nous avons trouvé deux images similaires à celle-ci, $t_1 = 40s$, $t_2 = 22s$. Dans la figure 15, nous avons trouvé trois images similaires à l'image originale (a). A la fin de cette section nous allons présenter une statistique de toutes les images de la base de données. Cette statistique donne la moyenne du rapport des taux de réussite des deux méthodes. Nous allons présenter un tableau qui contient : le nombre moyen d'images semblables R , le seuil t calculé à partir de R , la moyenne d'images semblables MIS calculée par la première méthode et la moyenne des taux de réussite MTR de toutes les images de la base. Le taux de réussite de chaque image est le nombre d'images semblables à l'image donnée en parcourant les centres sur le nombre d'images semblables en parcourant toute la base de données.



(a)



(b)



(c)

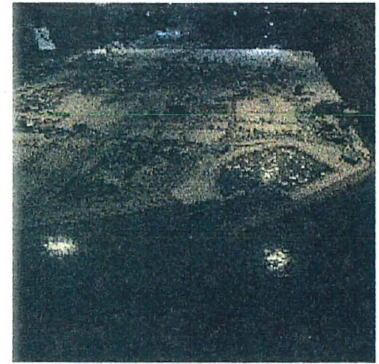
FIG. 13 – L'image originale est dans (a). Nous avons trouvé deux images similaires à celle-ci. Le nombre moyen d'images $R = 5$ et le seuil $t = 206.6$.



(a)



(c)

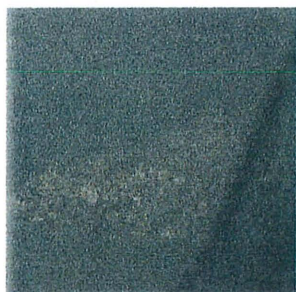


(d)

FIG. 14 – L'image originale est dans (a). Deux images similaires à cette image ont été identifiées. La première image (b) est la plus semblable à l'image originale, $R = 5$ et $t = 206.6$.



(a)



(b)



(c)



(d)

FIG. 15 – *Trois images similaires à l'image originale, $R = 5$ et $t = 206.6$.*

R	t	MIS	MTR
2	170.9	3.53	0.739176
5	206.6	7.48141	0.731655
10	221.9	11.0254	0.71066
15	237.2	15.5832	0.687177
20	247.4	19.6477	0.662246
25	257.6	24.1918	0.634916
30	262.7	26.8513	0.584611

TAB. 4 – Moyennes des taux de réussite, pour 511 images, pour chaque valeur de R donnée par l'utilisateur.

CONCLUSION

Dans ce mémoire, nous avons explicité la construction d'une base d'images indexée et ainsi que l'extraction d'images semblables à une image donnée, à partir de l'histogramme couleur.

Nous avons construit une base de données de 511 images couleur prises au Liban dans des environnements différents. Chaque image est représentée par trois histogrammes représentant les couleurs rouge, vert et bleu.

Dans la recherche des images semblables à l'image donnée, l'utilisateur choisit une image-requête et recherche dans la base des images disponibles celles qui ressemblent à cette image. Pour faire cette recherche, nous avons utilisé deux méthodes. Dans la première méthode, nous calculons la distance entre les histogrammes de l'image donnée et les histogrammes de la base, et nous extrayons les images dont la distance avec l'image donnée est inférieure ou égale à un certain seuil calculé. Ce seuil dépend du nombre moyen d'images donné par l'utilisateur. Cette méthode a conduit à des résultats satisfaisants ; cependant, dans le cas d'une grande base de données, le temps de recherche est long.

Pour réduire le temps de recherche, nous avons utilisé la deuxième méthode, qui est basée sur des réseaux neuronaux SOFM (Self-Organizing Feature-Mapping). Dans cette méthode, nous avons regroupé les histogrammes de la base de données dans des groupes distincts. Nous avons supposé que chaque groupe admet un centre significatif qui le représente. Nous comparons les histogrammes de l'image donnée avec les centres des groupes de la base de données au lieu de les comparer avec chacune des images de la base. Ensuite

nous trouvons le centre le plus proche de cette image. Enfin nous comparons cette image avec les images dont la distance par rapport à ce centre est inférieure ou égale à une certaine distance calculée.

Dans le chapitre I, nous avons implémenté des nouveaux algorithmes basés sur des formules mathématiques [1] et [2]. Ces algorithmes est pour indexer la base de données et construire un système de recherche des images. Dans le chapitre II, nous avons utilisé l'algorithme SOFM [12] et [13] pour améliorer ce système. Nous avons créé 5 programmes ayant en tout 2800 lignes codés en C++ sous UNIX.

La recherche d'images par similarité joue un rôle important dans plusieurs domaines. Il reste à améliorer notre travail pour pouvoir l'utiliser efficacement. Par exemple il est possible de construire un système complet capable de rechercher les images semblables à l'image donnée à partir des couleurs, textures et formes. Ce système pourrait indexer un grand nombre d'images couleur.

Bibliographie

- [1] Yihong Gong, Chua Hock & Guo Xiaoyi, Image Indexing and Retrieval Based on Color Histograms, Nanyang Technological University, Multimedia Tools and Applications, 2(2):133-156, 1996.
- [2] Markus Stricker & Michael Swain, The Capacity and the Sensitivity of Color Histogram Indexing, Technical Report 94-05, The University of Chicago, March 24, 1994.
- [3] Swain, Michael J. Interactive indexing into image databases, Technical Report 93-14, The University of Chicago, 1993.
- [4] Equitz, W. Hafner & J. al., Efficient color histogram indexing for quadratic form distance functions, IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(7):729-736, 1993.
- [5] Stricker, M. A. Color and geometry as cues for indexing Departement of Computer Science, Technical Report CS 92-22, The University of Chicago, November 1992.
- [6] Funt, B. V. & Finlayson, Color constant indexing, Simon Fraser University, Vancouver, Transactions on Pattern Analysis and Machine Intelligence, 17(5):522-529.
- [7] Swain, M. J. & Ballard, D. H. Color indexing Intern. Journal of computer vision, 7(1):11-32, 1991.
- [8] Engelson, S. P. & McDermott, D. V. Image signatures for place recognition and map construction, 1991.

- [9] Niblack & W. Barber, The QIBC project : Querying images by content using color, texture, and shape, 1993
- [10] Nadia Magnenat-Thalmann et Daniel Thalmann, Gestion de Fichiers et Bases de Données, Collection Systèmes d'Information, Genève, 1984.
- [11] Jean-Paul Guillois, Techniques de compression des images, Editions Hermès, Paris, 1996.
- [12] Kohonen T., Self-Organisation and Associative Memory, Second Edition, Springer Verlag, Berlin, 1984.
- [13] Paul Bourret et James Reggia, Réseaux Neuronaux, Editions Takenea, Toulouse, Novembre 1991.
- [14] Dayhoff J., Neural Network Architectures, Van Nostrand Reinhold, New York, 1990.
- [15] Van Lint, J. H., Introduction to Coding Theory, Second Edition Graduate Texts in Mathematics Springer Verlag, 1992.
- [16] André Nicoulin, Analyse d'images, Presses Polytechniques et Universitaires Romandes, Collection Meta, Lausanne, 1990.
- [17] Jacques E. Besançon, Vision par ordinateur, Editions Eyrolles, Paris, 1988.
- [18] M. N. Pons & J. P. Bellot, Visualisation des écoulements, des transferts et des structures, Institut National Polytechnique de Lorraine, Collection Récents Progrès en Génie des Procédés, Nancy, 1993.
- [19] Laurent Schwartz, Topologie Générale et Analyse Fonctionnelle, Collection Enseignement des Sciences, Paris, 1970.